

The POLY 88 Microcomputer System



If someone offered to sell you a computer to use in your home, you'd probably think he was nuts.

What on earth would you use it for? Where would you have room to put it? How could you afford to supply it with power, programs and peripheral equipment? And who would operate it for you?

The answers are surprisingly simple and practical. They come in the form of an extraordinary and unique personal tool, the POLY 88 microcomputer by PolyMorphic Systems.

Your entire family will be able to use and enjoy the POLY 88. It is a perfect introduction for children to the world of computers, stimulating them to use their intelligence and imagination. You can easily learn to live with the POLY 88—in fact it may become a highly valued member of the household.





Capabilities of the POLY 88 will impress you—It does far more than its compact size would suggest. The entertaining game and educational packages are but one side of it's personality. Versatile and powerful, the POLY 88 will assist you with many facets of personal finance, household routines, information storage and—what do you need? Set up your own dieting program. Analyze your energy usage. Create a space saving electronic file, accessible and useful. The possibilities are varied and far-reaching. One application may suggest many others.

The POLY 88 is a complete system from input to output. It works under home conditions, taking little space and using house current. And not only can you operate it, you can learn to program it in a short time—every aspect of the computer will be at your fingertips.

This brochure explains and demonstrates the elements of the POLY 88 system, a few of its applications, and shows an actual “run-through” of a specialized program. Please read through the brochure and take it along when you visit your POLY 88 feature dealer to operate the POLY 88.

The POLY 88 microcomputer system, for all of its abilities and potential, is a device which is perfectly suited for operation by any person—not just a computer expert. The only “language” you need to command is English!

Communication with the computer is accomplished through familiar devices. A TV monitor presents information in words and pictures, tables and graphs, economically or as fully detailed as you wish. A tape cassette recorder plays programs to the computer (or can record programs of your own invention). The standard typewriter keyboard is a convenient and universal method of directing your computer. All the elements of the system are designed for practicality and ease of use.

Actual operation of the computer is performed in a language called “BASIC”, which utilizes many English words to direct its functions. Using self-explanatory statements like PRINT, RUN, PLOT, and responses like YES or NO, you will be able to converse with your system in a brief time.

Please follow our set-up and demonstration in the following pages, and see how friendly a computer can be.

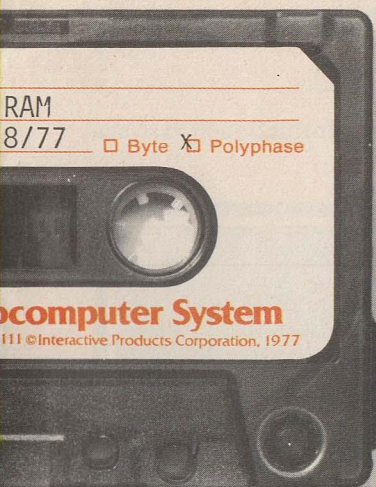
What is it like to have a computer in your home?

This "get acquainted" tour will give you the feel of using a POLY 88. The components set up compactly on a table, desk or shelf.

Have a seat in your favorite chair. Turn on the TV monitor and the POLY 88. The cursor, a small white rectangle, will appear on the screen.

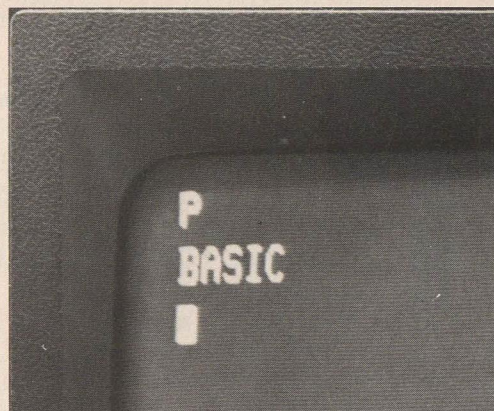
The POLY 88 is now ready to read a program from the tape cassette. Although the computer could be programmed directly from the keyboard using special "machine language" codes, these are not tailored for easy use by anyone. The language called "BASIC", on the other hand, simplifies communication with the computer by interpreting English word commands from the user. So the first step is to load "BASIC" into the computer.





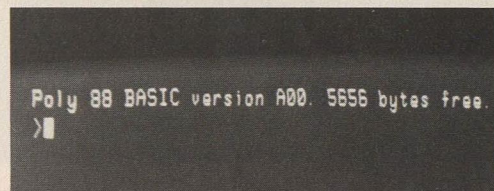
LOADING BASIC INTO A POLY 88

Put the cassette labeled "BASIC" into the cassette recorder. Type the letter **P** on the keyboard. This tells the computer that it will receive a program in our "POLY" format (much faster than the "Byte" standard, which can also be read by the POLY 88). Now type the word "**BASIC**". Then depress the carriage return key. The video display reads: ➡

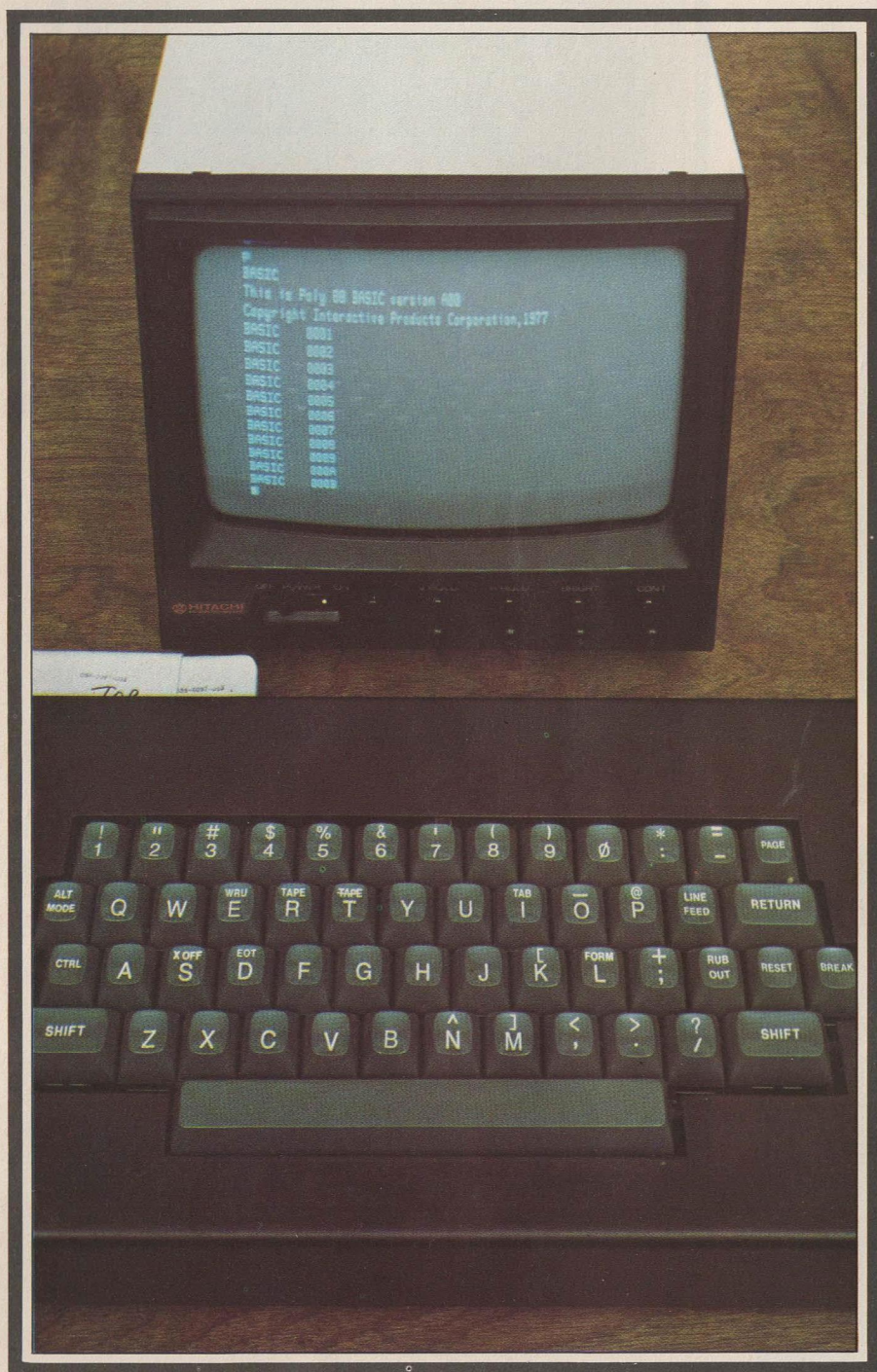


The computer is "listening" for a program called "BASIC", (the interpreter language). Start the cassette in the Play mode. A message will soon appear on the screen, followed by a series of numbers (one for each data block of 256 characters) as BASIC is loaded into the computer.

When loading is complete, the screen will clear and the computer will await your instructions. When you see a message at the top of the screen followed by a "prompt" symbol (`>`), you are ready to program in BASIC.



Why use the BASIC language? Because it accepts English words and phrases as commands. The command **LOAD**, for example, tells the computer to read and run a variety of application programs written in BASIC. These applications range from simple information storage to comprehensive financial and statistical analysis. Simply stated, BASIC programs make full use of the enormous computing power of the POLY 88, and put *that power* in the hands of people with no previous programming experience.

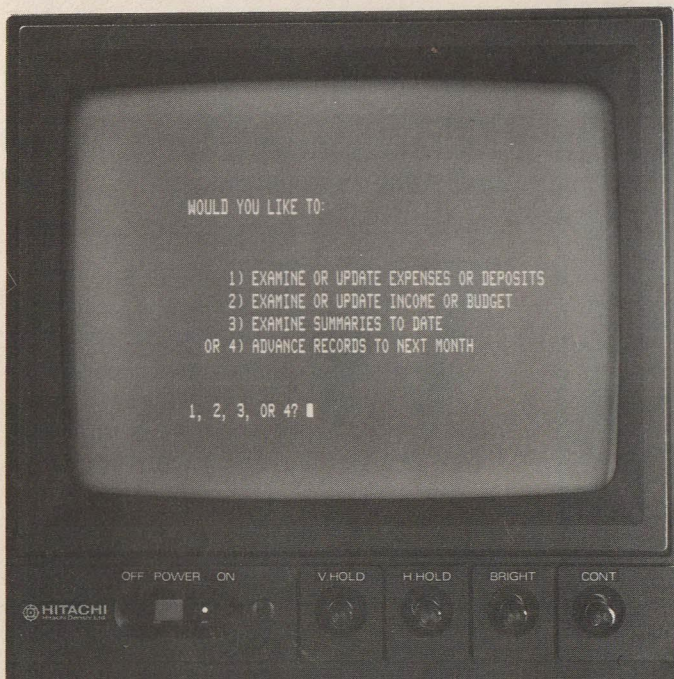


One such BASIC program is

CASHFLOW

The personal financial package which is used to write budgets, record transactions and reconcile checking accounts. To load CASHFLOW, type: **LOAD, CASHFLOW, P** and hit the carriage return key.

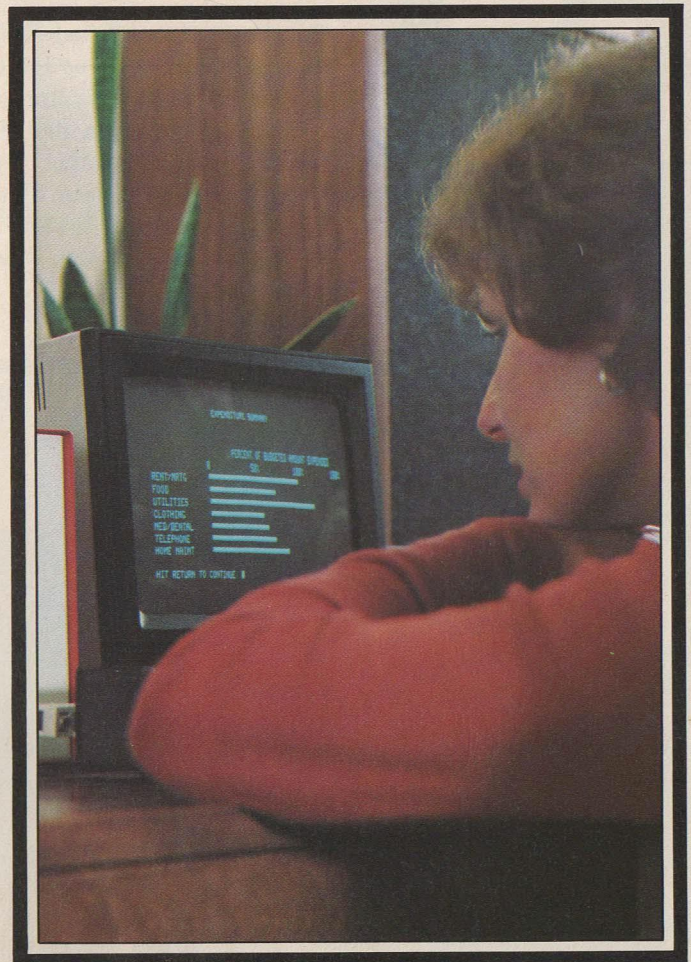
The screen will say "Working..." and wait for the tape input. The sequence is the same as for loading BASIC. Start the CASHFLOW tape in the play mode on the recorder. As it loads, the program will print on the screen. When finished, CASHFLOW will autoexecute. CASHFLOW will ask:



To see the financial summaries of a typical family, type **3** CASHFLOW responds:

ITEM	AMOUNT BUDGETED	AMOUNT EXPENDED	NET
RENT/MRTG	312.73	312.73	.00
FOOD	260.00	187.57	376.49
UTILITIES	60.00	71.97	-31.06
CLOTHING	120.00	68.50	57.00
MED/DENTAL	140.00	87.39	721.71
TELEPHONE	55.00	39.41	-71.47
HOME MAINT	60.00	53.43	31.07
TOTAL	\$1,007.73	\$821.00	\$1,083.74

Type **YES** (or simply "Y") to see the information in the form of a graph.



Hit a return to see the next set of budget items.

ITEM	AMOUNT BUDGETED	AMOUNT EXPENDED	NET
CAR LOAN 1	197.63	197.63	.00
CAR LOAN 2	102.87	102.87	.00
GAS, OIL	75.00	43.89	-51.07
CAR INS	44.79	.00	134.37
CAR MAINT	15.00	.00	45.00
REPAIRS	50.00	12.79	137.21
COMMUTING	40.00	33.50	-12.75
TOTAL	\$525.29	\$390.68	\$252.76

A "YES" or "NO" answer would allow you to see or bypass each remaining budget category (e.g., "FINANCIAL", "DISCRETION") or the deposit, withdrawal, income and expenditure summaries. For now, hit a return to end the summary listings, so you can see how a check or other transaction is entered.

Type 1 to see the

TRANSACTION ENTRY FORM

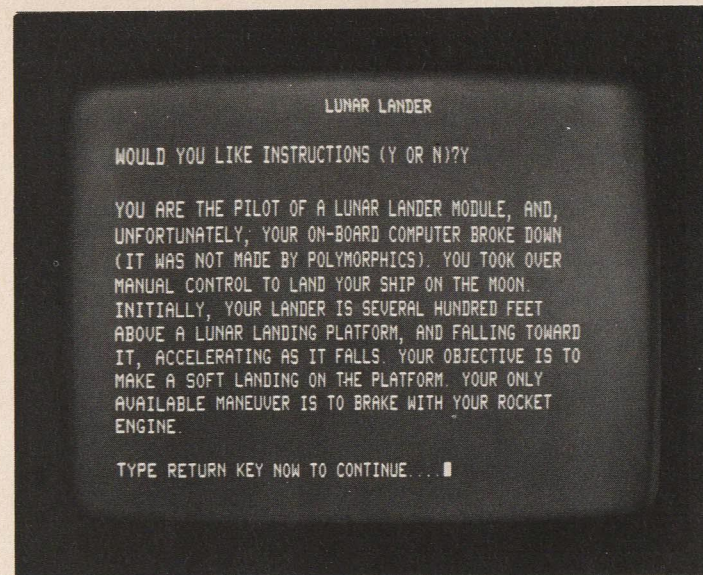


Using this form, you enter the type and amount of a new check, with automatic debiting of the corresponding budget account; or make any other transaction, including deposits, service charges, transfer of funds, etc. To review past transactions, scan by hitting the return key to go back one transaction, or by hitting the space bar to go forward one transaction.

After you have updated all your transactions, you would record the program on a new tape for use the following month, and save each old tape to form a complete financial record.

You don't have to use CASHFLOW to handle your finances—write your own program, tailored and customized to *your* circumstances.

But POLY 88 is not just dollars and cents. The whole family will enjoy fascinating animated computer games. For example, LANDER...



LANDER is a game program which tests your skill as a lunar astronaut. You must use your retro-rocket skillfully and efficiently to land safely on the moon.

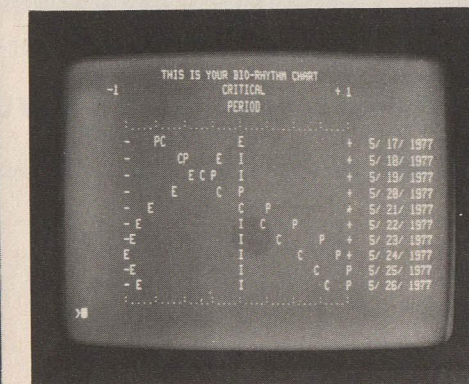
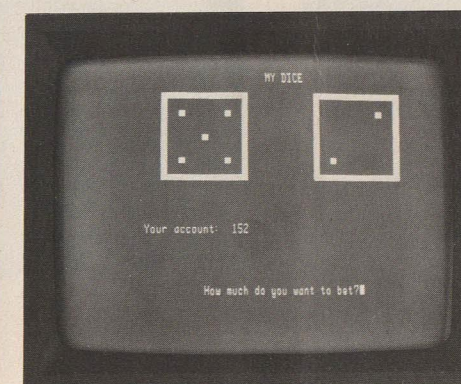
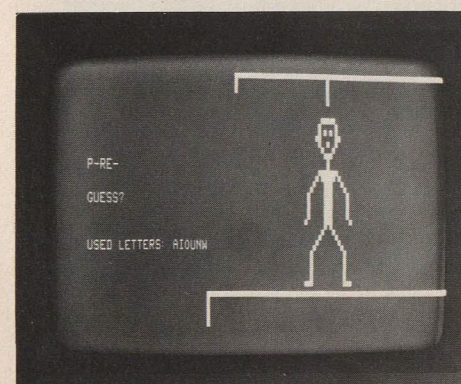
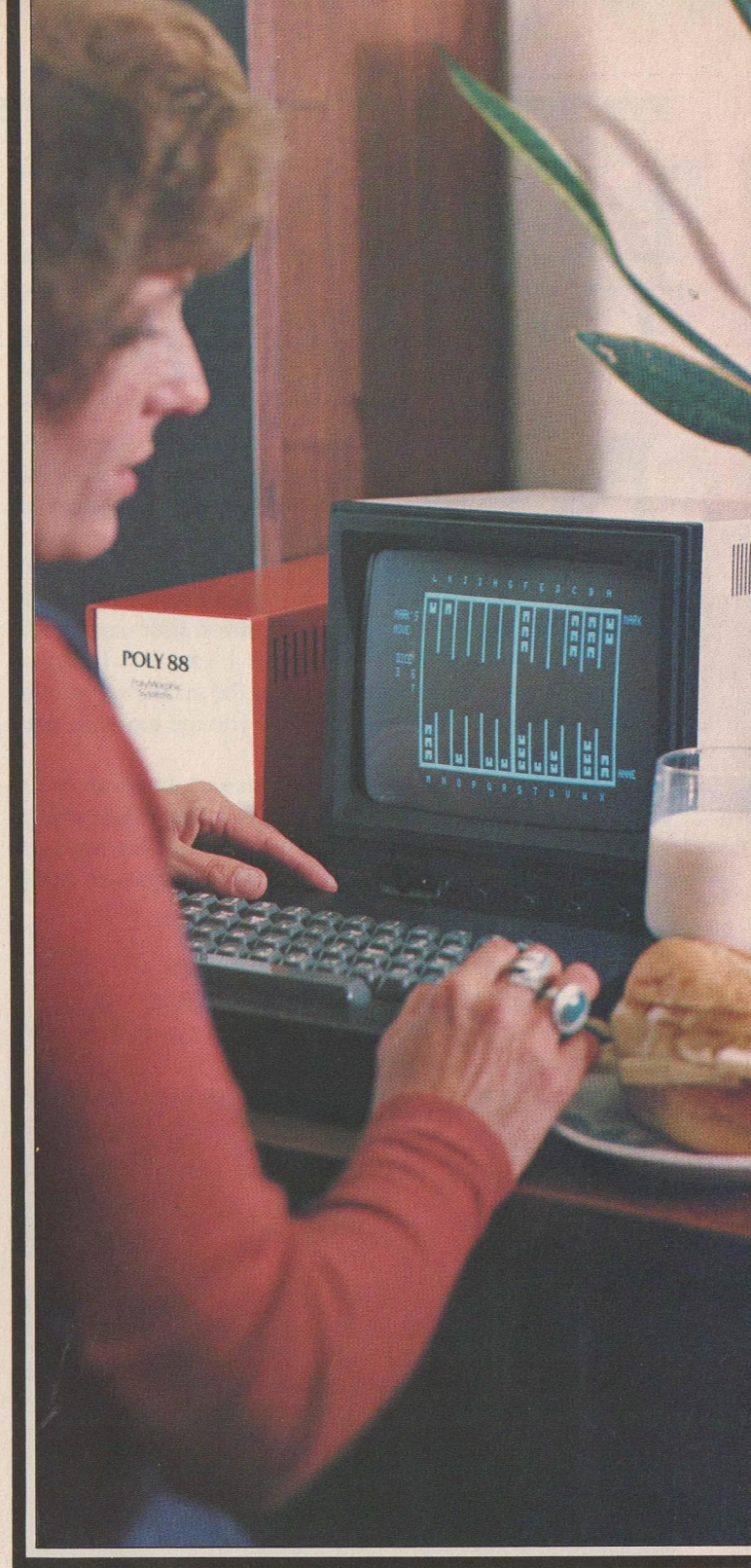
To load LANDER, follow the simple steps.

Type **LOAD, LANDER, P** hit the return key. The computer answers "Working..."; Play the LANDER tape. After the data blocks finish loading on the screen, the program will start. It will ask if you want instructions. Type **YES** and hit return. The game instructions will appear on the screen. To land safely, you must select and vary the rocket thrust, keeping an eye on altitude, speed and fuel consumption. Each game is slightly different. Miss the platform and you crash—and get a free critique of your flying skills from the computer! This game will give you hours of fun and excitement...

GAMES

...or sample our other challenging games, like BACKGAMMON...test your spelling ability under pressure with the HANG-MAN...break the bank shooting CRAPS (honest dice, no house odds)...

cheer home a derby winner with HORSE...get in touch with your biorhythms with BIOCYCLE...or any of our other fascinating games. They are easily used by the novice, and help develop your programming skills. For added enjoyment, devise your own games! There is nothing quite like the satisfaction of programming your own diversions, with personalized messages, varying difficulties, and differing strategies. The POLY 88 can be a very educational companion, growing with a child or jogging an adult's mental skills. It's easy with our BASIC language—so simple and straightforward that you will feel right at home with it in a few weeks.



LEARNING THE BASICS

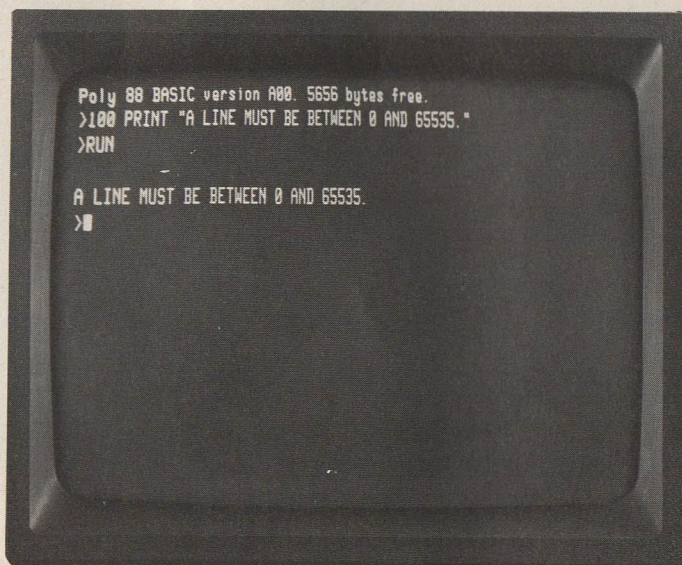
Now that you have seen the range of the POLY 88's abilities using our applications, you're ready to see what it can do in your hands. *What, you've never run a computer until right now?* That's all right; no experience is necessary to become proficient in BASIC. The commands are simple and direct. For example, type **SCRATCH** and hit return. You have just erased the last game or application in the machine (but you are still in the BASIC language), and you are ready to program.

A program is a series of statements and directions written in lines. To let the computer know you are programming, give it a line number (the number lets the computer index the information and makes it easier for you to refer to or repeat that line later).

For instance, type:

**100 PRINT "A LINE NUMBER
MUST BE BETWEEN 0 AND 65535."**

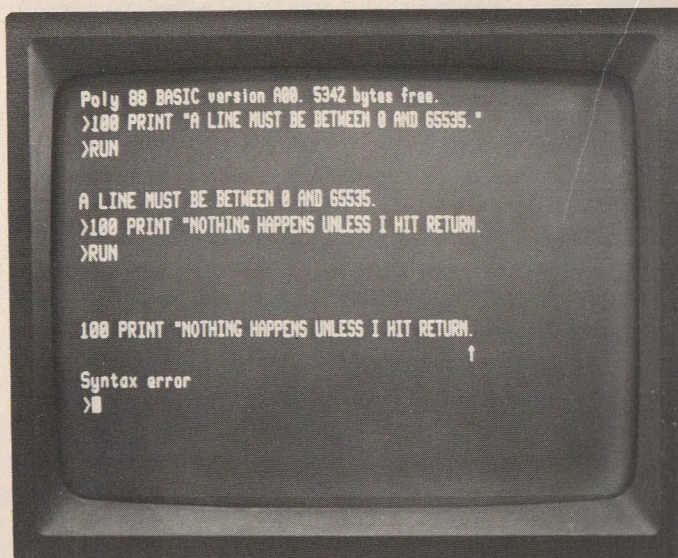
and hit return. Type **RUN** and hit return again. The video display shows:



You have just programmed the computer—to print a line that you specified. Try this one. Type:

**100 PRINT "NOTHING HAPPENS UNLESS
I HIT RETURN.**

Once again, hit return, type **RUN** and hit return again.



The computer does not understand this instruction because it was typed incorrectly. Compare it to your first program, and retype it as you think it should be. The computer will “prevent” you from making errors in this fashion by calling attention to the error (in this case, an error of syntax or language use). Suppose you catch your own error as you are typing the line? Try it. Type:

100 PRINT "USE THE DELTE KEY"

To correct, hit the key labeled **DELETE** until the cursor backs up over the error. Then type the correct version. Your finished line should read:

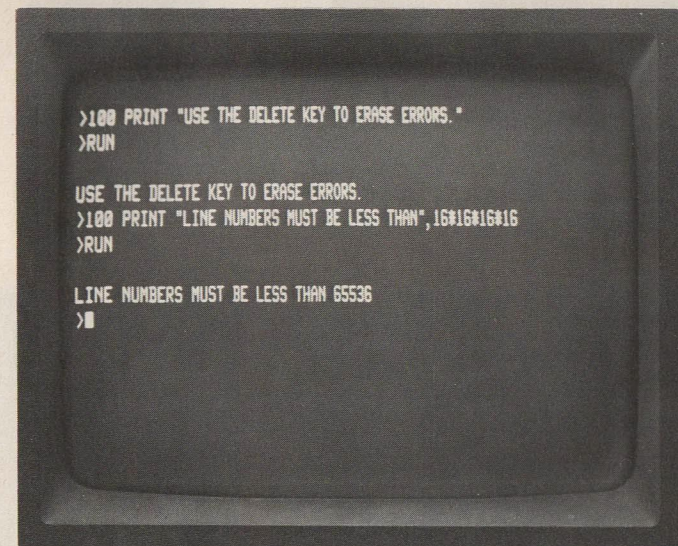
**100 PRINT "USE THE DELETE KEY
TO ERASE ERRORS."**

Hit return, type **RUN**, hit return again.

Congratulations! You are now experienced, with three programs to your credit. Now try something a bit different. Type:

**100 PRINT "LINE NUMBERS MUST BE
LESS THAN", 16*16*16*16**

Run the program (hit return, type RUN, hit return).



As you can see, it's done a multiplication problem. Now type **100 N=16*16*16*16** (hit return)
200 PRINT "LINE NUMBERS MUST BE LESS THAN", N

Run the program (return, RUN, return). This program adds a variable—that is, "N" represents the number, whatever you assign to it. The number assigned to "N" was set in line 100, and used in line 200. Any letter can be used to represent a number this way.

Now, put column headings in their places by typing:

**100 PRINT TAB (3), "DATE", TAB (15),
"TIME", TAB (34), "PLACE"**

Run the program.

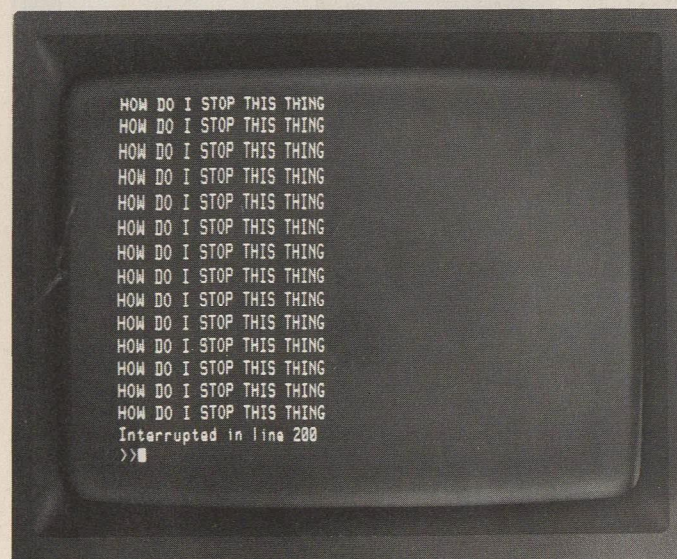
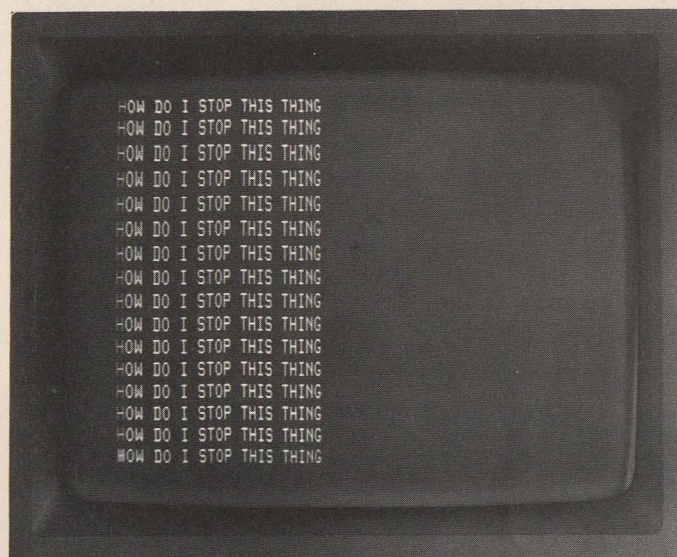
Typing the word TAB tells the computer how much to space over from the left margin before printing the word. Want to repeat a line more than one time? Just refer back to it in a later line. This two-line program does just that.

100 PRINT "HOW DO I STOP THIS THING?"

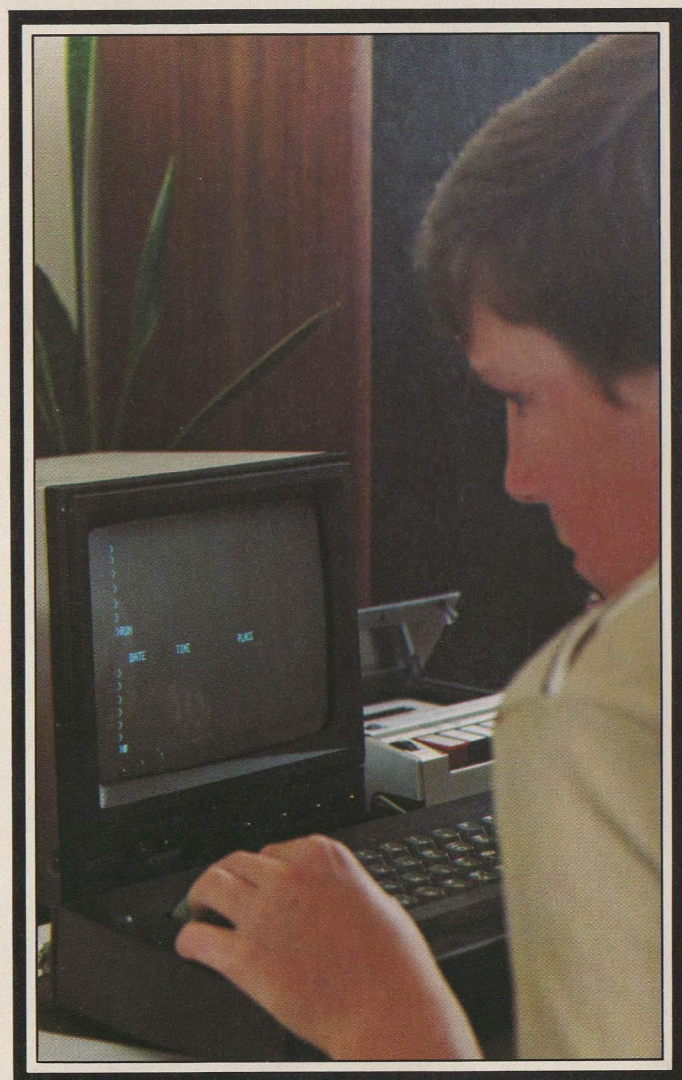
Hit return. Type:

200 GOTO 100

Run the program. Following your instructions, it will repeat itself... Hit a few keys. Now before you get desperate to stop it, hold down the key labeled CNTRL and hit Y. This will always stop a program, and return you to BASIC.



Try a few more programs. Get fancy... have the computer call you by name. When your name comes up, have it give you a title. May we suggest the title of "Programmer?"



To delete this last program,

Type SCRATCH

The last program you did simply returned the computer to a preceding line, printing it over and over. Suppose you want the computer to print something a specific number of times, then stop?

THE PROBLEM: Have the POLY 88 print a statement five times. **THE PROGRAM:**

**100 PRINT "THE POLY 88 CAN
COMMUNICATE IN BASIC."**

(this line tells the computer what to print)

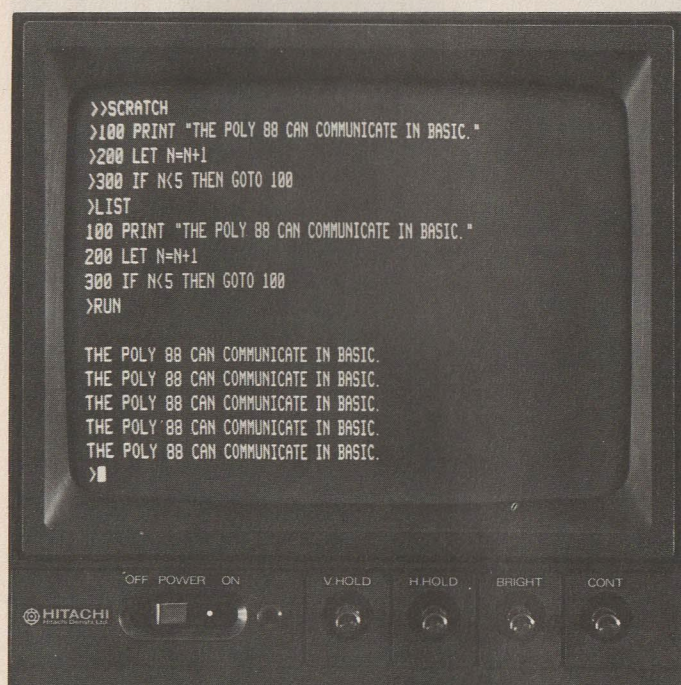
200 LET N=N+1

(N is the counter. Each succeeding time the computer goes through the program, the value of N will increase by 1)

300 IF N<5 THEN GOTO 100

(If the value of N is less than 5, return to line 100 and print the statement again.) NOTE: when N reaches 5, the computer will not return to line 100, but will go to a new line. Since there are no further lines, the program ends.

Once the program is typed in, hit return, type **LIST** and hit return again . . . to show the program, as you typed it, in its entirety. Now **RUN** the program.



Using the BASIC commands covered on the last few pages some very powerful programs can be developed. "The Manhattan Indian Problem", which follows, will demonstrate this, and will tell you once and for all whether New York was a good buy!



THE MANHATTAN INDIAN PROBLEM

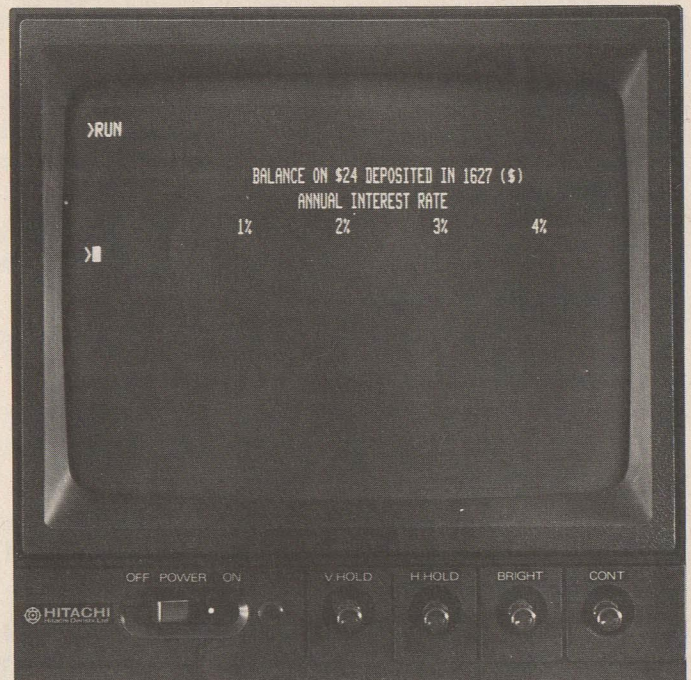
In 1627 Peter Minuit bought Manhattan Island from Chief Powhatan for approximately \$24.

THE PROBLEM: If the \$24 had been deposited in a savings account, what would be the value of the account? Calculate the answer for 50 year intervals from 1650 to 2000, using interest rates of 1%, 2%, 3%, and 4% compounded annually.

THE PROGRAM: The following is one way to solve the problem. The program has two parts, the first of which sets up a tabular display of the information. To ready the POLY 88, type SCRATCH and hit return.

Type the five lines below into the computer, then RUN the program. The screen should look like this: ➡

If an error is shown, compare the line with the corresponding line below. To correct it, retype and enter only that line.



100 PRINT TAB (20), "BALANCE ON \$24 DEPOSITED IN 1627 (\$)"

(this centers the title for the table)

150 PRINT TAB (26), "ANNUAL INTEREST RATE"

(this centers the title for the columns)

200 LET I=I+1

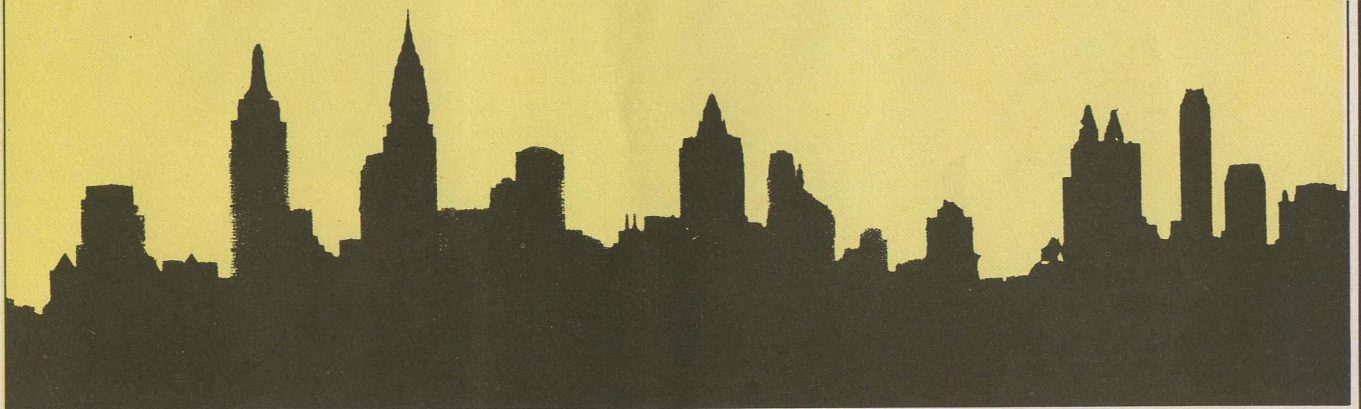
(this causes I to increase its value by 1 with each succeeding run-through of the program. First time through, I=0)

250 PRINT TAB (13*I+4), I, "%",

(this puts the proper number over each column one at a time. The comma after % will cause the next line bearing a PRINT instruction to print on the same line again and again as directed)

300 IF I < 4 THEN GOTO 200

(this sends the computer back to line 200 each time it gets to line 300, until all columns—4 of them—have a heading. When this is done—when I reaches 4—the program will continue to the next line)



The second part of the program is

THE CALCULATING SECTION

350 PRINT

(the cursor has been staying on the same line because of the comma at the end of line 300. This lone PRINT statement moves the cursor on to the next line.)

400 LET Y=1600

(the letter Y will represent, and count, the years. Remember, variables will start at zero unless otherwise assigned. Here we are assigning 1600 as the beginning year)

450 LET Y=Y+50

(each succeeding value of Y will be greater by 50, to provide our 50-year intervals)

500 LET I=0

(we used several values of I previously, up to I=4, to lay out the table in line 250. Now it is started again for calculating and formatting. If I=1, the program prints the first column; I=2, the second; and so forth)

550 PRINT Y,

(this prints the year in the left column. As in line 250, the comma after Y keeps the cursor on the same line, to print the appropriate value in each of the 4 columns)

600 LET I=I+1

(this line updates I by 1 (percent) on each succeeding run-through, to do each column in order)

650 PRINT TAB (13*I), 24*(1+I/100)^(Y-1627),

(this line does the calculation using the formula for compounding interest. It then prints the answers in the proper column) The symbol ^ is used for exponentiation (e.g., X^Y means X raised to the Y power)

700 IF I < 4 THEN GOTO 600

(this creates a "loop" between lines 600 and 700, to calculate and print the 4 values of I for each year—one per column)

750 PRINT

(this brings the cursor down to the next line (see 350))

800 IF Y < 2000 THEN GOTO 450

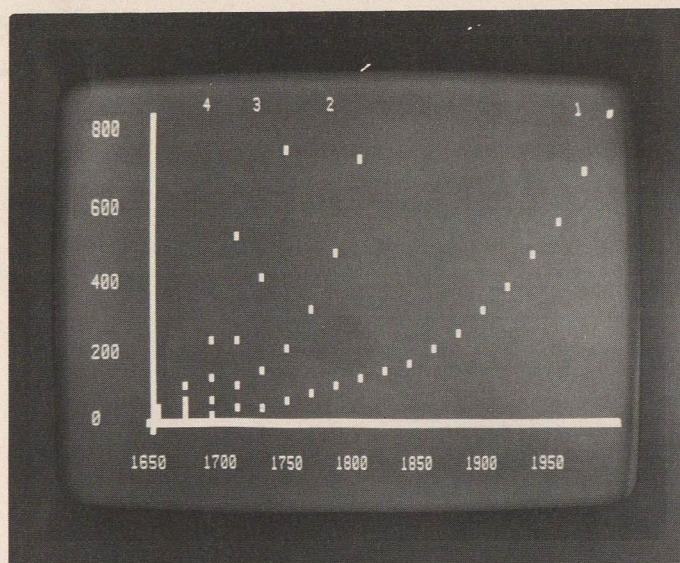
(the calculating loop between lines 600 and 700 does several interest problems for each year. This second loop between 450 and 800 causes the computer to continue calculations up to the year 2000. After Y reaches 2000—there being no further lines—the program stops.

Now RUN the program, and you will see:



The following program demonstrates further capabilities of BASIC—graphic representation. It uses some terms we have not covered — PLOT, FOR, NEXT, EXIT, for example—so we won't explain it in detail. It is presented to show that the full power of the POLY 88 can be exploited using the BASIC language.

*No other
microcomputer
has these graphics
capabilities.*



```

10 !CHRS(12)\ FOR D=800 TO 0 STEP -200\!!D\!\NEXT
20 FOR T=1 TO 7\!TAB(8*T-4), 1600+50*T,\NEXT\!CHRS(11)
30 FOR Y=6 TO 45\PLOT 15, Y, 1\NEXT
40 FOR X=14 TO 126\PLOT X, 7, 1\NEXT
50 IF I<4 THEN I=I+1 ELSE POKE 63552, 127
60 FOR X=0 TO 111 STEP 6\Y=1.08*(1+I/100)^(23+3.125*X)
70 IF Y<41 THEN PLOT X+16, Y+7, 1\NEXT
80 PLOT X, 47, 0\!!\EXIT 50
  
```

If you've looked all over the keyboard for \ , try "FORM" (shift L).

In only 8 lines, the screen is cleared, plotting axes with appropriate scales are drawn, and the results of the Manhattan table are plotted as a graph. Type this program in line for line, and see how the resulting graph looks. It is not necessary to understand how the program works to realize the great potential for presenting information in different ways.

You have demonstrated to yourself the power of BASIC to solve difficult, specialized problems with relative ease. Look how far you've come—and you've barely scratched the surface! Your POLY 88 dealer will be happy to show you other features of this outstanding computer, its simplicity and its reasonable price. You're probably dreaming up your own applications already—now make them a reality.

SPECIFICATIONS

Chassis

Power Supplies: +9V at 6A, +18 at 0.75, -18 at -25A (unregulated) per chassis.

Number of cards: 5

Compatibility: Accepts POLY 88—/IMSAI/Altair cards.

Bus Connector Type: Power switch, reset button, halt indicator and power indicator.

Cabinet Dimensions: 4¼" w X 6¾" h X 17" d.

Expandability: Up to 4 chassis may be plugged together.

Processor Card

Processor Type: 8080A

Clock: Crystal controlled 1.8432MHz.

Number of Instructions: 78 with 244 variations.

CPU Registers: 10

Addressing Modes: Direct, immediate, register, register indirect.

Addressing Range: 65,536 bytes of memory and 256 ports.

Interrupts: 8 level vectored.

Real Time Clock: 50/60 Hz (referenced to line frequency).

Bus Loading: Outputs drive 30 TTL loads, inputs are 1 TTL load or less

Power Requirements: +8 to 10V at 1.2 amps max., +16 to 20V at 200 mA. Max., -16 to 20V at 140 mA. max. (inc. 3 2708 EPROM).

RAM: 512 bytes 500 nsec. access time.

ROM: Sockets provided for 3K bytes of EPROM or (2708 type) ROM.

Serial I/O Port (option)

Baud Rate: Software controlled 12.5 to 9600 baud (asynchronous) 800 to 57,600 baud (synchronous).

Data Format: 5, 6, 7, or 8 bits with or without parity.

Ports: Can handle up to 2 minicards.

Power Requirements: +8 to 10V at 150 ma, -16 to 20V at 35 ma.

Video Terminal Interface

Character Font: 7 X 9 matrix alphanumeric and 6 element graphic characters.

Character Set: 96 character ASCII plus 32 special characters.

Number of Characters: 1024; 64 characters per line with 16 lines.

Graphic Resolution: 128 horizontal by 48 vertical.

Video Output: Composite video negative sync.

Required Video Bandwidth: 5.5 MHz.

Input Port: 8-bits plus positive or negative strobe.

Power Requirements: +8 to 10V at 1.25A typ., +16 to 20V at 30 ma max. -16 to 20V at 20 ma max., not including keyboard requirements.

Byte PolyPhase Cassette Interface

Baud Rate: 300, 600 (Byte); 1200, 2400 (Polyphase).

Connector (to recorder): Standard 25 pin "D" type female Mating connector type=DB-25p or equivalent.

Connector type (to CPU): 14 wire ribbon cable with DIP plug termination.

Power Requirements: +5±0.25 VDC at 150 ma max., -5±0.25 VDC at 20 ma max.

Input: Approximately 2Vp-p into 15 Ω

Motor Controls: (2) will switch up to 200 mA at up to 30V (negative ground).

8K RAM

Storage Capacity: 8192 X 8 bits

Memory Type: 91L02 or equivalent.

Access Time: 500 ns. max.

Address Decoding: Switch selectable in 8K increments.

Memory disable: Controlled by bus pin 67 (PHANTOM).

Power Requirement: +8 to +10 at 2.6A max. +2.2V 1.5A max. (standby).

Poly I/O Ideaboard

I/O Addressing: Blocks of four addresses.

Power Requirement (excluding user added components): +8 to 10V at 370 mA. max.

Firmware Monitor

Size: 1K bytes in 2708 type read-only memory ROM or compatible mask programmed ROM.

Operation: Fits in zero-addressed CPU ROM socket. Runs on system power-up or front panel reset.

Functions: Tape loader for Byte standard 300 baud or Polyphase 2400 baud encoded Polyformat absolute binary cassette files. Front panel simulator program produces register and memory display on system video display. Commands allow register, memory modification, program interrupt and single step or return to interrupted program.

Utilities: Routines for teletype simulated I/O on the memory mapped video display, character fetching from system keyboard, hexadecimal conversions, real time clock.

Features: "Wormholes" allow reassignment of I/O devices independent of program I/O handling. Video driver recognizes TAB, FORMFEED, BACKSPACE, LINE ERASE, VERTICAL TAB and CARRIAGE RETURN. Address of video display is reassignable. Interrupts handled through a vector table which allows reassignment of service routine addresses. Real time clock has 2 year delay capability with automatic execution of preset task on timeout.

11K BASIC by PolyMorphic Systems

Size: 11K bytes.

Scientific Functions: Sine, cosine, log, exponential, square root, random number, x to the y power.

Formatted Output

Multi-line Function Definition

String Manipulation and String Functions

Real-Time Clock

Point-Plotting on Video Display

Array Dimensions limited by Memory

Cassette Save, Load and Verify of Named Programs

Multiple Statements per Line

Renumber

Memory Load and Store

8080 Input and Output

If Then Else

Input Type-Ahead

Commands: RUN, LIST, SCR, CLEAR, REN, CONT, SAVE, LOAD, VERIFY.

Statements: LET, IF, THEN, ELSE, FOR, NEXT, GOTO, ON, EXIT, STOP, END, REM, READ, DATA, RESTORE, INPUT, GOSUB, RETURN, PRINT, POKE, OUT.

Built in Functions: FREE, ABS, SGN, INT, LEN, CHR\$, VAL, STR\$, ASC, SIN, COS, RND, LOG, TIME, WAIT, EXP, SORT, CALL, PEEK, INP, PLOT.

Printer Interface

Interface Standard: Full RS-232 or current loop (hardware selectable).

Required Input From CPU: TTL level serial data, clock, port select, and control

BAUD Rate: 57,600 max.

Output to CPU: TT level serial data, clock and control signals.

Connector Type (to serial device): EIA standard 25 pin female. Mating connector type + DB-25P or equivalent.

Connector Type (to CPU): 14 wire ribbon cable with DIP plug termination.

Dimensions: 2.1" X 4.0" (5.33 cm X 10.16 cm).

Storage Temperature: -25 to +85 C.

Operating Temperature: +5 to +55 C.

Humidity: 0 to 95% non-condensing.

Power Requirements: +5±0.25 VDC at 42 ma max. +12±0.6 VDC at 24 ma max., -12±0.6 VDC at 23 ma max.

**PolyMorphic
Systems**